

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-154115

(43)Date of publication of application : 08.06.1999

(51)Int.Cl.

G06F 12/08  
G06F 12/08  
G06F 15/163

(21)Application number : 10-235473

(71)Applicant : HEWLETT PACKARD CO <HP>

(22)Date of filing : 21.08.1998

(72)Inventor : KENNETH K SMITH  
LOREN P STALEY  
SORIN RAKOBOVICH

(30)Priority

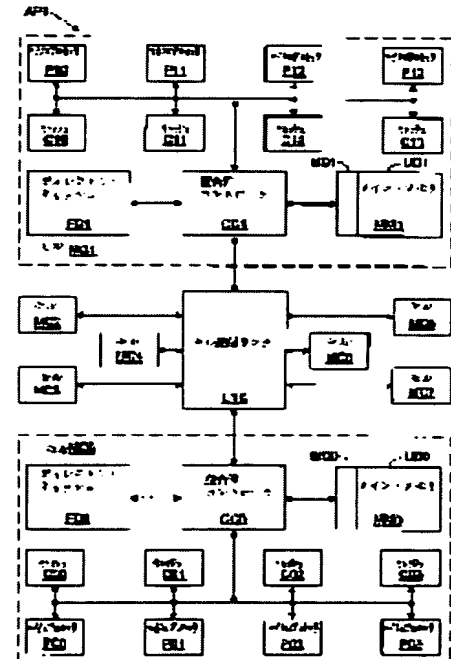
Priority number : 97 918209    Priority date : 25.08.1997    Priority country : US

## (54) DISTRIBUTED MEMORY MULTI-PROCESSOR SYSTEM

(57)Abstract:

**PROBLEM TO BE SOLVED:** To improve the processing ability of a distributed memory multi-processor system having cache matching ability by responding to a data request starting an expected recall based on information taken out from a quick matching directory before corresponding information is taken out from a main matching directory.

**SOLUTION:** The matching controller CC0 of a possessor cell MC0 starts the local inquiry of the cache, accesses the quick directory FD0 and starts the access of the main memory MM0. The matching controller CC0 judges whether quick directory data request the recall or not. When the recall is made, main directory data are compared with quick directory data. When directory data do not match correction measures are taken. It contains the restoration of a state changed in response to the start of the new recall and the expected recall.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or

application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-154115

(43) 公開日 平成11年(1999) 6月8日

(51) IntCl.<sup>6</sup>

G 0 6 F 12/08

識別記号

3 1 0

15/163

F I

G 0 6 F 12/08

15/16

E

3 1 0 B

3 2 0 K

審査請求 未請求 請求項の数1 OL (全 14 頁)

(21) 出願番号 特願平10-235473

(22) 出願日 平成10年(1998) 8月21日

(31) 優先権主張番号 9 1 8, 2 0 9

(32) 優先日 1997年 8月25日

(33) 優先権主張国 米国 (US)

(71) 出願人 398038580

ヒューレット・パカード・カンパニー

HEWLETT-PACKARD COMPANY

アメリカ合衆国カリフォルニア州パロアルト ハノーバー・ストリート 3000

(72) 発明者 ケネス・ケー・スミス

アメリカ合衆国83713アイダホ州ボイジー、ウエスト・ラダー・ドライブ 11776

(72) 発明者 ロレン・ビー・スティリィー

アメリカ合衆国95614カリフォルニア州クール、ウィンディ・コート 1810

(74) 代理人 弁理士 岡田 次生

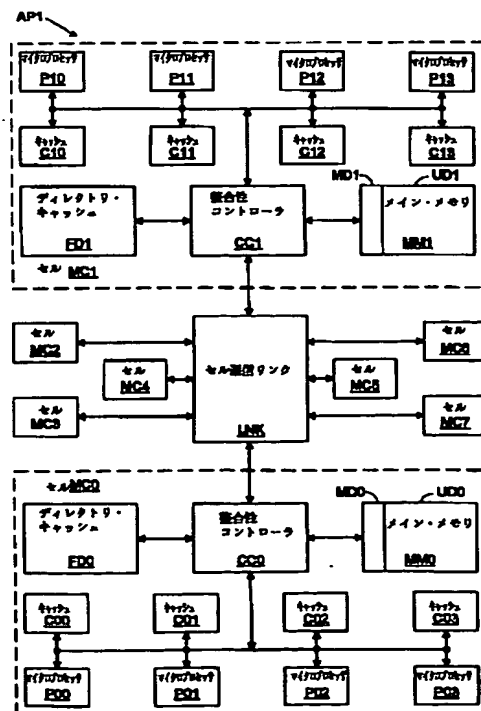
最終頁に続く

(54) 【発明の名称】 分散メモリ・マルチプロセッサ・システム

(57) 【要約】

【課題】 ディレクトリに基づくキャッシュ整合性を持つ分散メモリ・マルチプロセッサ・システムの処理性能を向上させる。

【解決手段】 メイン・メモリの整合性ディレクトリに加えて、キャッシュ記憶される迅速整合性ディレクトリを使用する。迅速ディレクトリは、メイン・ディレクトリ情報の一部のみをキャッシュ記憶する。データ要求に回答して、迅速ディレクトリ情報がメイン・ディレクトリ情報と並列してアクセスされる。迅速ディレクトリからの情報に基づいて、見込みリコールが発信される。引き続き受け取られるメイン・ディレクトリ情報によって、見込みリコールの有効性が判断され、無効であれば、決定性リコールが発信され、有効であれば、見込みリコールが継続する。見込みリコールはアクセス・タイムを減少させるので、見込みリコールの成功度合いに従って全般的システム性能が向上する。



## 【特許請求の範囲】

【請求項1】ディレクトリに基づくキャッシュ整合性を持つ分散メモリ・マルチプロセッサ・システムであって、

該システムは、相互に通信できるように接続され、複数のプロセッサ、複数のユーザ・データ・キャッシュ、複数のメイン・メモリ、複数の迅速整合性ディレクトリおよび複数の整合性コントローラを集合的に含む複数のメモリ・セルを備え、

上記複数メモリ・セルの各々が、上記プロセッサの少くとも1つ、上記キャッシュの少くとも1つ、上記整合性コントローラの1つ、上記メイン・メモリの1つおよび上記迅速整合性ディレクトリの1つを含み、

上記メイン・メモリの各々が、ユーザ・データを記憶し、整合性情報を表す1つのメイン整合性ディレクトリを含み、最大メイン・メモリ・アクセス時間を持ち、

上記迅速ディレクトリの各々が、それぞれのメイン整合性ディレクトリにおいて表される整合性情報のすべてではないが一部を表し、上記最大メイン・メモリ・アクセス時間より速い最速メモリ・アクセス時間を持ち、

上記整合性コントローラの各々が、そのそれぞれの迅速整合性ディレクトリおよびそのそれぞれのメイン整合性ディレクトリに並列的にアクセスして、対応する情報が上記メイン整合性ディレクトリから取り出される前に上記迅速整合性ディレクトリから取り出される情報に基づいて見込みリコールを始動させることによってデータ要求に応答する、

分散メモリ・マルチプロセッサ・システム。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、コンピュータ・システムに関するもので、特に、キャッシュ整合性分散メモリ・マルチプロセッサ・システムに関するものである。本発明の主な目的は、一層迅速な平均メモリ・アクセスを提供することにある。

## 【0002】

【従来の技術】現代の進歩の多くは、コンピュータ技術の急速な発展に関連している。すぐれた性能と豊富な機能に対する欲求が刺激される以上に強力な性能と豊富な機能を備えたコンピュータが急速に登場している。

【0003】コンピュータは、典型的には、1つまたは複数のプロセッサおよびメモリを含む。コンピュータ・メモリは、「ビット」と呼ばれる基本単位である2進数データの形式で情報を記憶する。メモリに記憶されるデータの大部分は、プログラム命令およびプログラム・データを含む「ユーザ・データ」である。プロセッサは、一度に多くのデータ・ビットを処理する。一度に取り扱われるビットの数が、組み込んでいるシステムに関するワード・サイズを定義する。初期のプロセッサは、一度に8ビットのワード(すなわち1バイト)を取り扱った。

現在32ビット・ワード・システムが普及していて、更に64ビット・ワード・システムがより広く使われるようになりつつある。

【0004】プロセッサは命令を実行するが、それはプログラム・データに対する演算を実行することを含む。マルチプロセッサ・システムは、単一プロセッサ・システムが順次にしか実行できない複数タスクを並列的に実行することによって一層高い処理性能を実現する。単一プロセッサ・システムと同様に、一部のマルチプロセッサ・システムは、単一化されたメイン・メモリをアドレスする。しかしながら、この場合、追加プロセッサの付加によって得られる利得は、複数プロセッサが単一化メモリへのアクセスを競合する際に発生する待ち時間によって相殺される。

【0005】メモリ競合を減らすため、メイン・メモリは少なくとも2つのメモリ・セルに分散される。各セルは、それ自身のメモリおよび1つ以上のプロセッサを含む。単一化メモリを想定するプログラムとの互換性を備えるため、各プロセッサは、(自身の)ローカル・メモリだけでなくセル通信リンク回路を経由して他のセルのメモリにもアクセスすることができる。ローカル・メモリのアクセスが遠隔メモリのアクセスより迅速ではあるが、すべてのメイン・メモリ・アクセスは、プロセッサ速度に比較すれば遅い。

【0006】キャッシュ機能は、メモリ・アクセスに付随する性能上の制限を改良することができる。キャッシュ機能は、メイン・メモリより小規模で速いキャッシュ・メモリにメイン・メモリの内容のサブセットを記憶する動作を伴う。キャッシュ内容がデータに対する要求を予測する確率を向上させるため種々の戦略が使用される。例えば、メモリ・アドレス空間における要求されたワードの近傍にあるデータは、要求されたワードと時間的に接近して要求される可能性が比較的高いので、大部分のキャッシュは、複数のワード行を取り出して記憶する。1つのキャッシュ行に記憶されるワードの数は、そのシステムに関する行サイズを定義する。例えば、1行が8ワード長である場合がある。

【0007】キャッシュは典型的にはメイン・メモリに比較して少ない行記憶位置を持つ。多くのメイン・メモリ行アドレスは、各キャッシュ位置と関連づけられる。従って、キャッシュ・データを所有するメイン・メモリ行アドレスをユニークに標示するタグが、各キャッシュ位置に、データと共に、記憶される。キャッシュにはいくつかのタイプがあるが、直接マップ・キャッシュが、各データ要求に対してただ1つのキャッシュ位置を調べるだけでよいので、最も迅速である。

【0008】単一プロセッサおよびマルチプロセッサ・システム両方において、キャッシュとメモリの間の「整合性」の保証という課題がある。例えば、プロセッサがキャッシュに記憶されたデータを修正するならば、その

修正はメイン・メモリに反映されなければならない。典型的には、データがキャッシュで修正される時間と、その修正がメイン・メモリに反映される時間の間にはなんらかの待ち時間が存在する。この待ち時間の間、メイン・メモリにおける未修整データは無効である。メイン・メモリ・データが無効である間それが読み取られないことを保証する処置が講じられなければならない。

【0009】マルチプロセッサ・システムにおける整合性の維持は、データが複数のキャッシュに並列的に記憶されることができるので、特に複雑である。1つのキャッシュにおいてデータの1つの複製が修正されるとすれば、他のキャッシュにおける対応するデータは無効となる。従って、どのキャッシュがどのデータを保有しているかを追跡し、他のキャッシュによるデータの複製の修正によってそのデータが無効となる時点を表示する手段が必要とされる。

【0010】典型的には、キャッシュに記憶されたデータを修正するためには「許可」が必要とされる。データが正確に1つのキャッシュに記憶されている場合のみ、その許可は与えられる。複数のキャッシュに記憶されたデータは、読み取り専用とみなされる。各キャッシュ行は、その行に記憶されているデータを修正する許可が与えられるか否かを標示する1つまたは複数の状態ビットを含むことができる。状態の正確な特性はシステムに依存しているが、典型的には、修正許可を標示するため「プライバシー」状態ビットが使用される。プライバシー・ビットが「プライベート」を標示すれば、1つのキャッシュだけがデータを保有し、それに関連するプロセッサはデータを修正する許可を与えられる。プライバシー・ビットが「パブリック」を標示すれば、複数のキャッシュがデータを保有することができるが、いかなるプロセッサもそれを修正することができない。

【0011】マルチプロセッサ・システムにおいて、ある1つのプロセッサがデータを読み取るかまたは修正することを望むとすれば、どのキャッシュがデータのコピーを持つか、また、データ修正の許可が与えられるか否かを決定する方法が存在しなければならない。「詮索機能」は、そのような決定を行うため複数のキャッシュの内容を調べる機能を含む。要求されたデータがローカル・キャッシュで検出されなければ、遠隔キャッシュが「詮索」される。別のプロセッサがデータを読み出すことができるように、プライベート・データをパブリックに変更することを要求するリコール(recall)が発信されるか、または、他のキャッシュが修正することができるようにいくつかのキャッシュのパブリック・データを無効化することを要求するリコールが発せられる。

【0012】詮索に必要とされる通信バンド幅は、詮索されるべきキャッシュの数に比例する以上に増加する。多数のプロセッサおよびキャッシュの場合、徹底的詮索は処理性能を悪化させる。このような理由のため、一部

の分散メモリ・マルチプロセッサ・システムは、セルの範囲内で詮索を行い、セル間の整合性を保つにはディレクトリに基づくキャッシュ整合性に依存する。

【0013】ディレクトリに基づくキャッシュ整合性を利用する分散メモリ・マルチプロセッサ・システムにおいては、各セルのメイン・メモリは、ディレクトリ・エントリをメモリの各行と関連づける。各ディレクトリ・エントリは、行をキャッシュに記憶するセルを識別し、そのデータ行がパブリックであるかプライベートであるかを識別する。セルの範囲内のどのキャッシュがデータを保有しているかを判断するため詮索機能を使用される。このように、各セルは、そのメイン・メモリに記憶されたデータのキャッシュ記憶されたコピーの位置を標示するディレクトリを含む。

【0014】例えば、8セル・システムにおいて、各ディレクトリ・エントリは、長さ9ビットである。セルの各々に関して、それぞれの「サイト(site)」ビットがそのセルが行のキャッシュ記憶されたコピーを含むか否かを標示する。9番目の「プライバシー」ビットは、そのデータがプライベートとしてまたはパブリックとして保有されているかを標示する。「プライベート」への状態の変化は、データを所有している(すなわちメイン・メモリに記憶している)セルに関わる整合性ディレクトリに先ず標示される。パブリックへの状態の変化は、先ずキャッシュに標示される。その他の場合、所与のデータ行に関して、キャッシュに標示されるプライバシー状態は、整合性ディレクトリにおいて標示されるプライバシー状態と一致する。整合性問題を回避するため、整合性ディレクトリにおける対応するプライバシー・ビットが「パブリック」を標示している限り、キャッシュ・プライバシー・ビットは「プライベート」を標示することを禁止される。

【0015】

【発明が解決しようとする課題】データがメイン・メモリから要求される時、リコールが必要か否かを決定するため関連整合性ディレクトリが検証されなければならない。リコールはメイン・メモリのアクセスの後でかつデータ要求が満たされる前に完了されなければならないので、一部のメモリ・アクセスはキャッシュのないシステムに比較して遅い。キャッシュはメイン・メモリ・アクセスの数を減らすので、性能全体は一般的には改善される。しかしながら、処理能力に対する飽くことを知らない需要を考慮すれば、なお一層の性能向上が求められている。

【0016】

【課題を解決するための手段】例えば大規模データベースのような一定タイプのユーザ・プログラムに関して、メイン・メモリ・アクセスに続くリコールは、少なくとも4つのセルを持つ分散メモリ・マルチプロセッサ・システムの性能を悪化させるほど頻繁に行われることが観

察されている。更に、その悪化はセルの数に比例する以上に大きい。このように、システムが大規模になればなる程、この性能悪化に対処する必要性が増大する。

【0017】考察された対策の1つは、メイン・メモリより速いメモリにメイン・ディレクトリを記憶することである。しかし、整合性ディレクトリ全体を記憶することのできるSRAMを備えるコストは、商業的に実行不可能のようである。従って、本発明は、メインの整合性ディレクトリの情報のすべてではなく一部を含む迅速整合性ディレクトリを提供する。

【0018】各迅速ディレクトリは、そのそれぞれのメイン・ディレクトリと同時にアクセスされる。情報はメイン・ディレクトリから取り出されることができるより迅速に迅速ディレクトリから取り出されることができるので、一部のリコールはメイン・メモリ・アクセスが完了する前に始動することができる。これらのリコールが正しく発信される度合いに応じて、リコールがメイン・メモリ・アクセスに続いて発信される頻度は減少する。このようにして、システム性能全体が向上する。迅速ディレクトリによって標示されないリコールは、メイン・ディレクトリが取り出された後発信される。

【0019】(例えばDRAM対SRAMのような)比較的迅速ではあるが高価なメモリ装置を経済的に利用するため、迅速ディレクトリは、メイン・ディレクトリより非常に小規模でなければならない。迅速ディレクトリが格納する行は、メイン・ディレクトリに比較して非常に浅い(すなわち行数が少ない)か、あるいは、狭い(すなわち行あたりのビット数が少ない)、またはその両方である。

【0020】迅速ディレクトリがメイン・ディレクトリに比較して浅い場合、それは、好ましくは、表される行が使用に従って変動するようなキャッシュとして運用される。キャッシュは典型的にはユーザ・データと共にタグを記憶することを必要とするが、本発明は、タグのないまたは不完全なタグを持つキャッシュを提供する。これは迅速ディレクトリを狭くすることに役立ち、所与のキャッシュ・メモリ容量に対して一層多くの行が表されることを可能にする。

【0021】不完全なタグをもつキャッシュは、大部分のアプリケーションに関しては考えられない。1つのタグ・ビットを省略することは、(キャッシュ記憶されているデータを2つのメイン・メモリ・アドレスへ変換するという)曖昧な対応付けを行うことになる。省略される付加タグ毎に曖昧さが倍増する。例えば $2^{20}$ (1M)の位置を持つ直接マップ方式のキャッシュが $2^{26}$ (64MB)の位置を持つメイン・メモリに対して使用されるとすれば、実際にデータをユニークに記憶しているメイン・メモリを標示するため6つのタグ・ビットが必要とされる。これらのタグ・ビットのいずれかが省略されると、2つのメイン・メモリ位置のどちらが表現されてい

るのか確かでなくなる。すべての6つのタグ・ビットが省略されると、64のメイン・メモリのうちのどれが表現されているのか不確実となる。

【0022】一方、タグ・ビットを省略することは、キャッシュ位置の数を2倍あるいは4倍にさえ増加させることができるので、曖昧さを半分または4分の1に減少させる。このように、64通りの曖昧さは、32通りまたは16通りの曖昧さに減少され得る。更に、キャッシング戦略がすべて効果をあげれば、すべてのアドレスが等しい確率で表現されるわけではない。要求アドレスがキャッシュで表されるアドレスである確率は、同じキャッシュ・インデックスに対応づけられるメモリ位置に関する表現の平均の確率より高い。

【0023】このように、要求アドレスがキャッシュで表されていると仮定することに対する正当な理由が存在する。本発明はこの仮定に基づいてリコールを始動する。迅速ディレクトリ・データが実際に要求アドレスに適用されることは不確実であるので、リコールは決定的ではなく予測的すなわち見込みである。要求アドレスが実際に迅速ディレクトリにおいて表されるアドレスであれば、リコールは成功であり、リコールがメイン・ディレクトリ・データに基づく場合に完了するよりも前に完了する。

【0024】不成功のリコールは、実質的には無害であり、エラーとなったユーザ・データは正しいユーザ・データと置き換えされない。一部のキャッシュ状態は、必要ではないのに、有効から無効へあるいはプライベートからパブリックへ変更される。このような不必要な状態変化は、無駄で潜在的に時間浪費的ではあるが、ユーザ・データ値に影響を及ぼさない。更に、一旦メイン・ディレクトリ・データが取り出されると、少なくとも一部のケースにおいて是正措置が講じられて状態がリセットされる。また、一旦メイン・ディレクトリ・データが取り出されると、正しいリコールを発することができる。いずれのケースでも、迅速ディレクトリなしの場合よりメモリ・アクセスが長いことはない。

【0025】不成功のリコールの主なコストは、通信トラフィックが付加されることである。このトラフィック付加は、主にアドレス経路に影響を及ぼす。アドレス経路はデータ経路に比較して活用度が少ない傾向があるので、典型的には余分なりコールのために利用できるいくつかの通信バンド幅が存在する。そうであっても、なお、余分なりコールを最小にするプロセスを実行することができる。

【0026】より多くのタグ・ビットの包含は、余分なりコールを減少させる1つの方法である。これがキャッシュ位置の減少の原因となるのであれば価値がないものと見なされるが、タグ・ビットの追加がそのような減少の誘因とならない状況が存在する。

【0027】迅速ディレクトリ・データは、リコールが

出される前に、整合性について(要求者セルまたは所有者セルからの)詮索データに対して検査される。取り出される迅速ディレクトリ・データが詮索データと整合していなければ、迅速ディレクトリ・データによって標示された見込みリコールは発信されない。

【0028】コストの点は別として、余分なりコールを減少させる好ましいアプローチは、迅速ディレクトリ位置の数を増加させることである。迅速ディレクトリ位置の数の増加は、余分なりコールの数を減少させるだけではなく、一層重要な点であるが、成功リコールの数を増加させる。極端な場合、ディレクトリ位置の数は、表現されるべきメイン・メモリ・アドレスの数に等しい(ディレクトリはキャッシュでなくなる)。この場合、表現されるアドレスに関する曖昧さはない。

【0029】迅速ディレクトリのビット単位のサイズが所与とすれば、記憶位置の数の増加は、位置あたりの記憶されるべきビットの数を減少させることを必要とする。特定の構成においては、メモリ装置は一層利用可能であり経済的であるので、エントリ幅単位の減少は他のものより価値がある。

【0030】エントリあたりのビット数減少は基本的には圧縮問題である。有効な見込みリコールを生成する際に役立つ情報は、あまり役立たないまたは冗長な情報を犠牲にして保持される。ローカルの詮索によって利用可能な情報は迅速ディレクトリに複製される必要はない。例えば、リコールを標示することができないすべての状況は、単一のコードによって表現されることができる。いくつかのケースにおいて、コンパクトなディレクトリ表現と見込みリコールの精度の間でトレードオフされる。

【0031】情報の相対的重要性は、ディレクトリ圧縮のみならずキャッシュ戦略にも密接に関係している。見込みリコールを行う目的のため、比較的古い価値のあるデータをより最近ではあるが価値の低いデータで書き換えることを拒否することは効果的かもしれない。例えば、プライベート・データがキャッシュされているセルを標示するエントリがアイドル・データを標示するデータ(これがキャッシュにあると仮定して)によって上書きされることを禁止するのは効果的キャッシュ戦略かもしれない。

【0032】多くのシステムにおいて、プライベート・データの読み取り試行は、共有された(パブリックでアイドルでない)データの書き込み試行に比較してきわめて普通である。迅速ディレクトリを備えていないシステム上での大幅な性能向上は、プライベート・データだけをキャッシュすることによって得ることができる。プライベート・データは、パブリック・データに比較して少ないビットで表されるので(8ビットの場合3ビット対8ビット)、キャッシュの良好な候補である。

【0033】区分化された迅速ディレクトリは、状態ビ

ットを記憶することなく2つ以上の状態を取り扱うために使用することができる。例えば、迅速ディレクトリは、プライベート・データのための「プライベート」セクションおよびパブリック・データのための「パブリック」セクションを含むことができる。データが記憶されているセクションが状態を決定するので、状態ビットは必要とされない。一層多くのセクションを使用して、キャッシュ位置あたりの記憶されるビット数を減少させることができる。

- 10 【0034】区分化されたディレクトリの重要な利点は、各セクションが同じ次元を持つ必要がないということである。プライベート・データがリコールを出すため一層役立つ傾向があるので、一層多くの位置をプライベート・セクションに割り当てることができる。位置あたりのビットをもっと多く使用することによって、パブリック・データはそのデータをキャッシュしているセルを一層完全に指定することができる。例えば、8セル・システムに関して、3サイト・ビットをプライベート・セクションのために使用し、8サイト・ビットをパブリック・セクションのために使用することができる。16セル・システムに関して、4サイト・ビットをプライベート・セクションのために使用し、8サイト・ビットをパブリック・セクションのために使用することができる。この場合、パブリック・データの見込みリコールはペアでセルに発信される。

- 30 【0035】区分化されたキャッシュでは、各セクションが各データ要求に対して一般的に調べられ、一方、区分化されてないキャッシュについてはただ1つの位置だけが調べられるだけでよい。2つのセクションからの情報が相違すれば、2セットのリコールが発信される。

- 40 【0036】迅速ディレクトリに記憶される情報の不完全な性質のため、一部の迅速ディレクトリ・データは、他の迅速ディレクトリ・データより役に立つことがある。例えば、ユーザ・データがプライベートに保持されていることを標示しユーザ・データがプライベートに保持されている位置を標示する迅速ディレクトリ・データを使用して単一の見込みリコールを発信することができる。データがアイドルであることを標示する迅速ディレクトリ・データは見込みリコールを発生させないので、性能強化とならない。従って、本発明は、データの効用が考慮される迅速ディレクトリ更新アルゴリズムを提供する。例えば、例え時間的に一層近くても、データがアイドルであることを標示しているデータはデータがプライベートに保持されていることを標示するデータを置き換えない。換言すれば、ある迅速ディレクトリを他のものと置き換えるべきかどうかの決定は、それぞれの迅速ディレクトリ位置において表されるデータが要求されたとすれば発信されるであろう見込みリコールの特性に部分的に基づく。この戦略は、コンテンツに対して中立である従来技術のキャッシュ更新アルゴリズムと異なる。

【0037】読み取り要求を取り扱う方法は、読み取り要求を出すステップ、それが要求者セル内で充足されることができるか検査するステップ、および充足されない場合要求を所有者セルに転送するステップを含む。所有者セルは並列的にローカル詮索を実行し、迅速ディレクトリにアクセスして、メイン・ディレクトリのアクセスを開始する。詮索データおよび迅速ディレクトリ・データは、リコールが標示されているか否か判断するため、相互に関係付けされる。データがメイン・ディレクトリから取り出される前にリコールは開始する。

【0038】一旦メイン・ディレクトリ・データが利用可能となれば、迅速ディレクトリから取り出される整合性データを確認または非整合を確認することができる。非整合が確認されれば、始動されたいかなるリコールも中止または反転される。その場合、決定性リコールが始動され、アクセス・タイムは迅速ディレクトリがない場合と同じとなる。次に、データは要求者プロセッサに提供され、状態が更新され、迅速ディレクトリはそのコンテンツに基づく更新アルゴリズムに基づいて更新される。迅速ディレクトリ・データ整合性が確認される場合、アクセス・タイムは向上する。書き込み要求も同じように取り扱われるが、但し、リコールは、1つ以上のセルが要求あたりリコールに従属するようにパブリック・データをプライベートにすることを含む。

【0039】本発明は、メイン・ディレクトリ・アクセスに続くリコールの頻度を経済的に減少させる。かくして、コンピュータ性能は一般的に改善される。好ましいタグなしキャッシュ実施形態に関する限り、不正確な見込みリコールの数を最小にするためのアプローチはいくつかある。ディレクトリ情報の圧縮は、装置キャッシュ・サイズあたりのディレクトリ位置が一層多数になることを可能にし、見込みリコール成功の確率を増加させ、不正確な見込みリコールの頻度を減少させる。

【0040】課題を解決する1つの手段として、本発明は、ディレクトリに基づくキャッシュ整合性を持つ分散メモリ・マルチプロセッサ・システムを提供する。該システムは、相互に通信できるように接続され、複数のプロセッサ、複数のユーザ・データ・キャッシュ、複数のメイン・メモリ、複数の迅速整合性ディレクトリおよび複数の整合性コントローラを集散的に含む複数のメモリ・セルを備え、上記複数のメモリ・セルの各々が、上記プロセッサの少なくとも1つ、上記キャッシュの少なくとも1つ、上記整合性コントローラの1つ、上記メイン・メモリの1つおよび上記迅速整合性ディレクトリの1つを含み、上記メイン・メモリの各々が、ユーザ・データを記憶し、整合性情報を表す1つのメイン整合性ディレクトリを含み、最大メイン・メモリ・アクセス時間を持ち、上記迅速ディレクトリの各々が、それぞれのメイン整合性ディレクトリにおいて表される整合性情報のすべてではないが一部を表し、上記最大メイン・メモリ・アクセ

ス時間より速い最速メモリ・アクセス時間を持ち、上記整合性コントローラの各々が、そのそれぞれの迅速整合性ディレクトリおよびそのそれぞれのメイン整合性ディレクトリに並列的にアクセスして、対応する情報が上記メイン整合性ディレクトリから取り出される前に上記迅速整合性ディレクトリから取り出される情報に基づいて見込みリコールを開始させることによってデータ要求に応答する。

【0041】

10 【発明の実施の形態】本発明に従って、ディレクトリに基づくキャッシュ整合性を備える分散メモリ・マルチプロセッサ・システムAP1は、図1に示されるように、セル通信リンクLNKを経由して相互通信可能なように接続された8つのメモリ・セルMC0-MC7を含む。メモリ・セルMC0は、4つのプロセッサP00-P03、4つのユーザ・データ・キャッシュC00-C03、メイン・メモリMM0、迅速整合性ディレクトリFD0および整合性コントローラCC0を含む。同様に、メモリ・セルMC1は、4つのプロセッサP10-P13、4つのユーザ・データ・キャッシュC10-C13、メイン・メモリMM1、迅速整合性ディレクトリFD1および整合性コントローラCC1を含む。メモリ・セルMC2-MC7は、メモリ・セルMC0およびMC1と本質的に同じものである。

【0042】メモリ・セルはすべて本質的に同一であるので、メモリ・セルMC0の詳細な記述は、残りのメモリ・セルMC1-MC7の構造および機能を定義するためには十分である。ユーザ・データは、メイン・メモリMM0のユーザ・データ・セクションUD0に記憶され、64ビット・プロセッサP00などのアクセスに備える。メイン・メモリは、4GB(ギガバイト)=256MW(メガワード)の容量を持っているので、システムAP1は、セルMC0-MC7に均等に分散された合計32GB=4MWのメイン・メモリを持つ。

【0043】キャッシュC00-C03は、メモリ・アクセスのために必要とされる平均時間を顕著に減少させる。1ワードがプロセッサP00によってメイン・メモリMM0からフェッチされる(すなわち取り出される)時、それはキャッシュC00に記憶され、それに加えて、要求されたワードに隣接する7つのワードがフェッチされ、要求されたワードとともに記憶される。従って、8ワードがフェッチされキャッシュ記憶位置に記憶される。システムAP1に関する行の長さは、8×64ビット・ワードすなわち512ビットである。

【0044】キャッシュC00の速度は、その比較的小さい容量と速いメモリの使用に起因している。キャッシュC00は、4MB=256KL(キロ行)SRAMモジュールである。そのようなメモリは、メモリMM0として4GB=256ML容量を提供する複数モジュールDRAM装置より非常に迅速にアクセスされる。



【0045】キャッシュC00の動作はプロセッサP00にとって透過的である。プロセッサP00によるデータ要求は、システムAP1の40億以上のメイン・メモリ位置を識別する32ビットのメイン・メモリ・ワード・アドレスを含む。キャッシュC00は、このアドレスをインターセプトして最下位3ビットを除去することによってそのアドレスを29ビットの行アドレスに変換する。この29ビット行アドレスが整合性コントローラC00に転送され、キャッシュC01-C03が詮索(すなわち照合)される。要求が所有者セルに転送されなければならぬならば、整合性コントローラは、最上位3ビットをメモリ・セルIDに復号して、要求が満たされるように適切なセルに残りの26ビット行アドレスを転送する。

【0046】キャッシュC00は、最下位行アドレス18ビットをキャッシュC00のキャッシュ位置を識別するため使用される256KL個のインデックスの1つと合致させる。次に、29ビット行アドレスのうちの残りの最上位11ビットは、選択されたキャッシュ位置に記憶される11ビット・タグと比較される。「ヒット」すなわちタグが合致する場合、最下位ワード・アドレス3ビットを使用して、プロセッサP00へ転送するためそのアドレス位置に記憶されている8データ・ワードのうちの要求された1つを選択する。「ミス」すなわちタグが合致しない場合、最終的にメイン・メモリからフェッチされる行が、選択されたキャッシュ位置のデータの行を上書きし、その位置のタグがプロセスの中で更新される。最後に、要求されたワードがキャッシュ行からプロセッサP00へ転送される。

【0047】整合性は、キャッシュC00に記憶されるユーザ・データの各行について3状態ビットを含むこと

#### データ状態および関連措置

	パブリック		プライベート	
	共有	非共有	クリーン	ダーティ
ローカル読み取り	Yes	Yes	Yes	Yes
ローカル修正	No	No	No	Yes
遠隔読み取り/キャッシュ	Yes/Yes	Yes/Yes*	Yes/No	No
遠隔修正	No	No	No	No

\*遠隔読み取りの際状態は「パブリック共有」に変更

【0050】共有ビットがない場合、すべての「パブリック」データは「共有」として扱われ、すべての「プライベート」データは「ダーティ」として扱われる。プライベート・データの遠隔読み取りはメイン・メモリが更新されデータ状態がパブリックに変更されることを必要とする。パブリック・データの修正は、データが他のキャッシュからリコールされデータがプライベートとされることを必要とする。

【0051】「共有」ビットは、実質的に、なにがしかの性能上の利点を提供する2つの中間状態を定義する。共有でパブリックなデータと相違して、非共有でパブリ

によって実施される。図2に示されるように、各ユーザ・データ・キャッシュは、11タグ・ビット、3状態ビットおよび512ユーザ・データ・ビットを各キャッシュ位置に記憶する(各位置はパリティなどのエラー訂正ビットも記憶することができるが、本発明の目的から無視することができる)。第1の状態ビットは、キャッシュ位置に記憶された行の有効または無効を標示する「有効性」ビットである。第2の状態ビットは、キャッシュ位置に記憶されたデータがパブリックかプライベートかを標示する「プライバシー」ビットである。

【0048】第3の「共有」ビットの意味は、プライバシー・ビットの値に依存する。プライバシー・ビットが、データはパブリックであると標示すれば、共有ビットは、データが別のセルにキャッシュ記憶されているか否かを標示する。共有に関する情報は、データを所有しているセルによって提供される。プライバシー・ビットがデータはプライベートであると標示すれば、共有ビットは、データが「(ダーティ(汚れている))」か「クリーン(きれい)」かを標示する。適切なプログラム命令にตอบสนองして関連するプロセッサがこの状態を設定する。プライベートでクリーンなデータは、所有者プロセッサによる読み取りは可能であるが、修正はできない。プライベートでクリーンなデータは、別のプロセッサによる読み取りは可能であるが、キャッシュはできない。セルが保持するプライベートでクリーンなデータは、データを修正するためリコール手段に頼ることなくその状態をプライベートで汚れた状態に変換することができる。次の表1は4つのデータ状態を要約して示している。

【0049】

【表1】

40 ックなデータは、リコールなしでプライベートにされることができる。プライベートなデータと相違して、非共有でパブリックなデータは、ローカル・プロセッサの許可なしに遠隔プロセッサによって読み取られキャッシュされることができる(この場合状態はパブリック共有に変更される)。ダーティでプライベートなデータと相違して、クリーンなプライベートなデータは、状態変更なしに遠隔プロセッサによって読み取られることができる。パブリックなデータと相違して、クリーンでプライベートなデータは、(ダーティ状態に変更後)容易に修正されることができるように遠隔キャッシュ記憶から保

護されることができる。

【0052】システムの初期化とともに、すべてのキャッシュ有効性ビットは無効に設定される。データ要求があると、要求されたアドレスによってインデックスを付けられたキャッシュ位置での有効性ビットが無効に設定されていれば、タグ・ビットの値に関係なくミスが標示される。データがキャッシュ記憶位置に入れられる時、有効性ビットは「有効」に設定される。有効ビットは、以下に記述されるように、リコールに応答して「無効」に設定されることができる。

【0053】プロセッサは、そのキャッシュにおけるいかなる有効データも読み取ることができる。しかしながら、プロセッサはそのキャッシュがプライベートとして保持するデータだけを修正することができる。プロセッサがパブリックとして保持するデータを修正する必要がある、データは先ずプライベートにされなければならない。プロセッサが関連キャッシュにないデータを修正する必要がある、データはそのキャッシュにプライベートとして入れられなければならない。データが別のキャッシュによって使用中であれば、データがプライベートにされる前にそのキャッシュからリコールされなければならない。

【0054】同じセルの他のプロセッサと関連するキャッシュの中の要求されたデータのコピーを特定するため詮索機能が使用される。プロセッサP10が、自身がパブリックとして保有するデータを修正することを必要とする場合、整合性コントローラCC1は、ローカル・キャッシュP11-P13におけるすべてのコピーのリコールを実行するため詮索機能を使用する。リコールは、プライベートとして保有されているコピーができるだけ早くパブリックに変更されることおよびパブリック・コピーが無効にされることを要求するはたらきを持つ。仕掛かり中のデータ・コピーがなければ、データのプライベート・コピーがプロセッサP10に提供されるか、あるいは、そのパブリック・コピーがプライベートにされる。そこで、プロセッサP10はデータのそのプライベート・コピーを修正することができる。

【0055】システムAP1におけるセル間整合性はディレクトリに基づくものである。要求がセル内部で満たされなければ、その要求は、要求されたデータを所有しているセルに関わる整合性コントローラに転送される。例えば、プロセッサP10が、メイン・メモリMM0の範囲内のアドレスを明らかにするとすれば、セルMC0が要求されたデータを所有する。整合性コントローラCC0は、要求されたデータ・システムのコピーの幅を検出する。この検索に必要な情報は、ユーザ・データと共に行単位で記憶されるメイン・ディレクトリの中に維持される。図2に示されるように、メイン・メモリMM0の各行は2つの状態ビットおよび8つのサイト・ビットを記憶する。8サイト・ビットは、各セルがその行のコ

ピーを保有しているか否かを各セル毎に標示する。第1の状態ビットは、データがパブリックまたはプライベートいずれとして保持されているかを標示し、第2のキャッシュ状態ビットに対応する。

【0056】第2の「共有」メイン・ディレクトリ状態ビットは、一般的に、キャッシュにおける「共有」状態ビットに対応する。キャッシュとは相違して、メイン・ディレクトリはキャッシュされていないデータを保有することができる。本発明の好ましい実施形態は、キャッシュされていないデータを単に非共有のパブリック・データの特例とみなす。このように、メイン・ディレクトリにおいては、共有ビットが、データが多くとも1セルによってキャッシュされている場合と少なくとも2セルにキャッシュされている場合とを区別することができる。メイン・メモリの共有ビットは、キャッシュの共有ビットと同じ方法で、プライベートでダーティなものとプライベートでクリーンなものを区別する。

【0057】代替的实施形態では、第2のメイン・ディレクトリ状態ビットが、「パブリックでアイドル」と「パブリックでキャッシュされた」状態を区別する。どれかのキャッシュによって保持されているパブリック・データは、「パブリックでキャッシュされた」ものである。キャッシュされてないデータは、「アイドル」でパブリックである。要求されたデータがアイドルであれば、要求はサイト・ビットに関係なくまたいかなるリコールを発信することなく即刻満たされることができる。(不正確な)迅速ディレクトリ情報に基づいて出されたいかなる見込みリコールも、サイト・ビットを参照せずに中止することができる。好ましい実施形態は、メイン・ディレクトリ状態ビットとキャッシュ状態ビットの一層密接な対応関係を維持する。

【0058】キャッシュとメイン・ディレクトリのプライバシーおよび共有状態ビットは、概ね対応するが、但し移行は同時に起こらない。パブリックからプライベートへの移行は、先ずメイン・ディレクトリにおいて標示され、プライベートからパブリックへの移行は、データをプライベートとして保有するキャッシュにおいて先ず標示される。(このように、キャッシュは、メイン・ディレクトリおよびシステムの残りがパブリックとして扱うデータを「プライベート」として扱うことはできない)。プライベートでダーティな状態とプライベートでクリーンな状態の間の移行は先ずキャッシュで標示され、一方、パブリックで共有の状態とパブリックで非共有の状態の間の移行は先ずメイン・ディレクトリで標示される。

【0059】メイン・ディレクトリMM0におけるデータ状態ビットから、整合性コントローラCC0は、システムAP1のどのセルが要求されたデータのコピーを保有しているかまたそのデータがパブリックとしてあるいはプライベートとして保有されているか判断することが

できる。このようにして、リコールは識別されたセルに向けられることができる。リコールを受け取る複数セルの範囲内で、詮索機能を通してデータの位置が定められる。

【0060】迅速ディレクトリFD0は、見込みリコールが始動されることを可能にする。ユーザ・データ・キャッシュC00などと同様に、迅速ディレクトリFD0は、メイン・メモリに比較して小規模(4Mb×4=16Mb=2MB)であるが、速い(DRAMでなくSRAM)。迅速ディレクトリFD0は、ユーザ・データ情報を記憶する代わりに、対応する整合性ディレクトリ情報を記憶するが、メイン・メモリMM0のメイン・ディレクトリMD0に記憶される整合性ディレクトリ情報ほど特定のではない。

【0061】迅速ディレクトリFD0に記憶されるディレクトリ・データは、図2に示されているように、モードAとモードBの間の設置場所での構成可能な選択に依存している。モードAは、2Mの4ビットキャッシュ・ディレクトリを提供する。1プライベート・ビットを記憶するため位置あたり1ビットが使用され、残りの3ビットは、セルMC0-MC7のうちの1つのIDを符号化する。2Mのキャッシュ・ディレクトリ位置は、64Mのメイン・メモリ位置をカバーし、各キャッシュ・ディレクトリ位置は、その位置に直接対応づけられる32メイン・メモリのうちの多くとも1つのアドレスに関するディレクトリ情報を記憶することができる。

【0062】モードAでは、迅速ディレクトリFD0は、タグのないキャッシュであり、図2に示されるように、モードAでは迅速ディレクトリFD0によってタグ・ビットは記憶されない。この結果、32の可能なメイン・メモリ・アドレスのうちのどれがそれぞれの迅速ディレクトリ位置によって表されるかを決定する直接の方法がない。その代わりに、要求されたアドレスによってポイントされるインデックスにおける迅速ディレクトリ情報がそのアドレスに適用されると仮定され、それに応じてリコールが始動される。以下に明記される場合を除いて、迅速ディレクトリFD0のあらゆるアクセスはヒットにつながるが、提供されるデータが間違ったメイン・メモリ行アドレスに対するものである可能性はある。

【0063】間違った迅速ディレクトリ情報に基づいて出されたりコールは、ユーザ・データ・エラーにならない。誤って導かれたセルMC0がセルMC2からのデータを要求すると、セルMC2は単に要求されたデータを見つけないだけである。メモリMM0からの真のディレクトリ情報が取得され次第いかなるエラーも訂正されることができる。間違ったりコールの負荷は、主に余分なセル間アドレス・トラフィックである。このトラフィックは、所有者セルとキャッシュにおけるコピーの仮定された保有者の間の通信に限定される。

【0064】タグ・データを省略する利点は、節約され

たメモリ容量を、ディレクトリ・キャッシュにおけるメモリの一層多くの行を表すことに捧げることができる点である。完全なタグを備えたとすれば、メイン・メモリ・アドレスの半分も使用することができなくなる。これは、成功リコールの数を減少させ、性能向上を阻害する。

【0065】単一の状態ビットが、キャッシュに保持されるデータがパブリックかプライベートかを標示する。それがプライベートであれば、キャッシュにおいて1つのセルだけがデータのコピーを持つことができる。そのセルをユニークに識別する上で、ディレクトリ・キャッシュにおける3つのサイト・ビットで十分である。データがパブリックであれば、0、1、2、3またはそれ以上のセルがそれを保持する可能性がある。第1の近似方法として、モードA符号化は、パブリック・データのコピーの1つの保有者だけを識別することができる。

【0066】キャッシュ・コントローラCC0は、セルMC0に関する整合性データを詮索機能によって決定する。従って、所有者セル・キャッシュにおけるデータの存在に関するディレクトリ情報は必要でない。これは、基本ディレクトリ・キャッシュ符号化方式の有益な修正を可能にする。

【0067】より具体的にいえば、2つの自己参照4ビット・コードは特定のケースに関して解放される。現在のケースにおいては、要求されたデータが所有者セルによってパブリックにキャッシュされていることを標示するコード0000を使用して、ディレクトリ・キャッシュ情報に基づいていかなるリコールも発信されるべきではないどのようなケースをも標示する。これは、書き込まれていないディレクトリ・キャッシュ位置、パブリックであるがアイドル(すなわちどのキャッシュにも記憶されていない)データ、所有者セルだけによってキャッシュされているデータ、および、所有者セル以外の少なくとも2つのセルによって共有されているデータをカバーする。

【0068】これはコード1000を残す。このコードは、要求されたデータが他の目的のため所有者セルによってプライベートとしてキャッシュされていることを標示する。例えば、このコードを使用して、所有者以外の例えば6、7という最小限の数のセルによって共有されるデータに対するグローバルなリコールを始動することができる。これは多数のトラフィックを生成するが、この発信はまれであり、メイン・ディレクトリ情報が所有者整合性コントローラによって取得され次第、これらリコールを中止することができる。

【0069】非自己参照コードは、所有者セルのローカルの詮索によって収集される情報に照らして解釈されなければならない。例えば、コード0001は、要求されたデータのコピーがセルMC1によってキャッシュされていることを標示する。所有者セルの詮索機能は、ま

た、要求されたデータがセルMC0によってキャッシュされているか否か判断する。いずれにせよ、要求者セルが所有者セルである場合(例えばプロセッサP00がメモリMM0に記憶されているデータを要求する場合)、リコールは必要とされない。迅速ディレクトリFD0に適用されるキャッシュ戦略において、非自己参照コードは、自己参照コードによって置き換えられない。これは、役立つ情報がそれほど役に立たない情報によって上書きされるのを防止し、平均して性能利得を達成する。

【0070】整合性コントローラCC0は、間違ったりリコールの数を減少させるため、要求者セルおよび所有者セルからの詮索データに対して迅速ディレクトリ・データを検査することができる。このため、要求者セルは、所有者整合性コントローラに適切な詮索データを提供する。このように、迅速ディレクトリFD0が要求されたデータがセルMC2によってプライベートにキャッシュされていることを標示するが、セルMC1が、データの共有コピーを持つことをセルMC0に通知する場合、整合性コントローラCC0は、そのディレクトリ情報が同じ下位ビットを持つ別のメモリ位置に対するものであったことを認識する。このようにして、有効なディレクトリ情報がメイン・メモリから取得されるまで、リコールは発信されない。同様に、ローカル詮索が、所有者セルがプライベートとしてデータをキャッシュしていると判断すれば、リコールは発信されない。

【0071】モードBでは、迅速ディレクトリFD0は、2つの相互に排他的で独立してインデックス付けされたセクションに分割される。それらセクションの一方が、プライベート・データに関するディレクトリ情報を保有し、他方が、所有者セル以外のセルによって使用されるパブリック・データに関するディレクトリ情報を保有する。状態ビットを必要としないので、プライベート・セクションは3つのサイト・ビットだけを必要とする。しかしながら、コンピュータは2の累乗を効率的に活用するので、4番目のビットは1ビットの「部分的タグ」として使用される。この1ビット・タグ位置における値は、完全なタグの最下位ビットであるものと一致する。言い換えると、それはディレクトリ・キャッシュにインデックス付けするために使用されない最下位ビットである。タグ・ビットと一致しないアドレスに対する間違ったりリコールを防止するため、不完全なタグが使用される。

【0072】パブリック・セクションは、自己参照コード00000001が所有者セルによる唯一のキャッシュではなくむしろ空のディレクトリ・キャッシュ位置を示すということを条件として、基本的にはメイン・ディレクトリの8つのサイト・ビットを複製する。符号化されないサイト・ビットを使用することによって、パブリック・セクションは、真のヒットの場合うまく目標を定められた見込みリコールが発信されることを可能にす

る。モードBにおける迅速ディレクトリFD0に関する限り、各セクションは512KBの容量を持つ。プライベート・セクションは、この容量を1Mのディレクトリ位置に割り当てるが、パブリック・セクションは512Kのディレクトリ位置を表現するためにこの容量を割り当てる。ほとんどの場合共有されたデータの書き込みに比較してプライベート・データの読み取りは一層頻繁に要求されるので、この割り当ては合理的である。

【0073】データ要求を取り扱う基本的方法M1の流れ図が図3に示されている。ステップS1において、プロセッサP11がメイン・メモリMM0に記憶されているデータの読み取り要求を発信する。ステップS2において、要求がローカルに満たされるか否かを判断するため、要求者セルMC1のキャッシュC10-C13が検証される。まず、関連キャッシュC11が検査される。ヒットすれば、要求はローカルで満たされる。ミスの場合、要求は要求者整合性コントローラCC1に委ねられる。整合性コントローラCC1はローカル詮索を始動する一方要求を所有者セルMC0に委ねる。詮索の結果ヒットすれば、要求はローカルに満たされることができ、データが別のプロセッサ例えばプロセッサP12によってプライベートに保持されていれば、整合性コントローラは、要求が満たすためデータをパブリックとすように要求する。ローカル詮索がミスする場合に限り、所有者セルMC0の参画が要求される。

【0074】ステップS3において、所有者セルMC0の整合性コントローラCC0は、そのキャッシュのローカル詮索を始動し、迅速ディレクトリFD0へアクセスし、メイン・メモリMM0のアクセスを始動する。整合性コントローラCC0は、迅速ディレクトリ・データがリコールを要求するか否か、また、ディレクトリ・キャッシュ・データがローカル詮索結果と一致しているか否か判断する。ディレクトリ・データが詮索結果と一致しリコールが標示されていれば、ステップS4において、リコールが始動される。

【0075】ステップS5において、整合性コントローラはメイン・メモリ・ディレクトリ情報を受ける。リコールが発信されていれば、整合性コントローラはメイン・ディレクトリ・データを迅速ディレクトリ・データと比較する。ディレクトリ・データが整合していなければ、訂正措置がとられる。これは新しいリコールの始動および見込みリコールに応答して変更された状態の復元を含む。

【0076】リコール・プロセスが完了すると、ステップS6において、要求されたデータが要求者セルMC1、整合性コントローラCC1、キャッシュC11およびプロセッサP11へ転送される。キャッシュC11、迅速ディレクトリFD0およびメイン・メモリMM0の整合性ディレクトリは必要に応じて更新される。ここで方法M1は終了する。

【0077】書き込み動作は、読み取り動作と同様である。システムAP1において、書き込み要求は、読み取り-修正-書き込み(read-modify-write)命令として実施される。書き込み動作の場合、データはローカルに見つけられることができるが、なおその他のセルがアクセスされる必要があるかもしれない。これは、データを共有しているセルからのデータのリコールが要求されることができるようデータがパブリックで共有状態である場合である。要求者セルがデータのコピーを持つという事実は、所有者セルへ伝えられることができる。所有者整合性コントローラは、それ自身のセルからの詮索情報とともにこの情報を使用して、ディレクトリ・キャッシュ・データの有効性を検査し、余分なりコールを排除することができる。

【0078】迅速ディレクトリの2方向区分化という前述の形態は、いかなる数のセクションにも拡張することができる。セクションの数の各2倍が、ディレクトリに記憶される必要があるビットの数の1ビットの減少を提供する。例えば、8セクション迅速ディレクトリは、8セルの各々に対して1つのセクションを含む。各ディレクトリ位置は、幅2ビットとすることができる。2ビットは、例えば、1)キャッシュされてない、2)キャッシュされプライベート、3)キャッシュされパブリックで非共有、および4)キャッシュされパブリックで共有、という4状態を識別することができる。共有および非共有データの区別を検証目的のため使用して、余分なりコールの数を減少させることができる。1つのセクションは、自己参照的であり、特別の目的のため排除されたり使用されたりする。

【0079】また、迅速ディレクトリに位置あたり1ビットだけを記憶することも可能である。そのビットは、そのビットを保有するセクションに対応するセルに対してリコールが発信されるべきか否かを標示する。リコールは、汎用目的とすることができる。例えば、共有またはプライベートであるデータを無効にするものとリコールを解釈するようにすることができる。

【0080】代替的形態として、キャッシュにおけるデータの状態に基づいて単一の信号が解釈されるようにすることができる。リコールは、例えば、「データがパブリックとして保持されていなければそれを無効化せよ；データがプライベートとして保持されていればそれをパブリックにせよ」ということを意味することもできる。成功リコールは、以前はパブリックであったデータの読み取りおよび書き込み動作を急がせる。以前はプライベートであったデータの書き込み動作は、メイン・ディレクトリ情報が取り出され次第別のリコールを必要とする。しかしながら、データがすでにパブリックであったので、時間はいくらか節約されるであろう。

【0081】プライベート・データをパブリックに投機的に変換することが見込みリコールに基づくあまりに過

激な措置であれば、リコールは、プライベートでダーティなデータをプライベートでクリーンなデータへ変換するものとして解釈されることができる。そのようにすれば、少なくとも、データが要求者セルに転送される必要があれば、メイン・メモリはそのデータの最新のコピーを持つであろう。

【0082】明らかに、迅速ディレクトリに記憶されるデータを制限する多くの可能な方法がある。前述の記述は、状態ビットおよびタグ・ビットを数の上でどのように減少させることができるか、あるいはもっと極端に言えば迅速ディレクトリからどのように排除するか教示している。それに加えて、サイト・ビットの数は、符号化およびセルのグループ化の両方またはいずれかによって減少させることができる。本発明の上述の実施形態に対する種々のバリエーションおよび変更は可能である。

【0083】本発明には、例として次のような実施形態が含まれる。

(1) ディレクトリに基づくキャッシュ整合性を持つ分散メモリ・マルチプロセッサ・システムであって、該システムは、相互に通信できるように接続され、複数のプロセッサ、複数のユーザ・データ・キャッシュ、複数のメイン・メモリ、複数の迅速整合性ディレクトリおよび複数の整合性コントローラを集成的に含む複数のメモリ・セルを備え、上記複数のメモリ・セルの各々が、上記プロセッサの少なくとも1つ、上記キャッシュの少なくとも1つ、上記整合性コントローラの1つ、上記メイン・メモリの1つおよび上記迅速整合性ディレクトリの1つを含み、上記メイン・メモリの各々が、ユーザ・データを記憶し、整合性情報を表す1つのメイン整合性ディレクトリを含み、最大メイン・メモリ・アクセス時間を持ち、上記迅速ディレクトリの各々が、それぞれのメイン整合性ディレクトリにおいて表される整合性情報のすべてではないが一部を表し、上記最大メイン・メモリ・アクセス時間より速い最速メモリ・アクセス時間を持ち、上記整合性コントローラの各々が、そのそれぞれの迅速整合性ディレクトリおよびそのそれぞれのメイン整合性ディレクトリに並列的にアクセスして、対応する情報が上記メイン整合性ディレクトリから取り出される前に上記迅速整合性ディレクトリから取り出される情報に基づいて見込みリコールを始動させることによってデータ要求に応答する、分散メモリ・マルチプロセッサ・システム。

【0084】(2) 上記迅速整合性ディレクトリがキャッシュである、上記(1)に記載のシステム。

(3) 上記迅速整合性ディレクトリがタグなしキャッシュである、上記(2)に記載のシステム。

(4) 上記迅速整合性ディレクトリが不完全なタグを記憶する、上記(2)に記載のシステム。

(5) 上記迅速整合性ディレクトリが状態情報のないサイト情報を記憶する、上記(2)に記載のシステム。

(6) 上記迅速整合性ディレクトリから取り出された情

報の有効性を検査して見込みリコールを発信すべきか否かを決定するためローカル詮索データが使用される、上記(2)に記載のシステム。

(7) キャッシュされるべきディレクトリ・データの情報値を考慮に入れるキャッシュ記憶アルゴリズムに従って上記迅速整合性ディレクトリが更新される、上記

(2)に記載のシステム。

【0085】(8) 分散メモリ・マルチプロセッサ・システムにおけるプロセッサからの読み取り要求を処理する方法であって、プロセッサが読み取り要求を出すステップ(a)と、b) 要求プロセッサを含むセルの内部で上記要求を満たすステップ(b)と、上記ステップb)で要求を満たすことができない場合以下のステップに進み、所有者セルの迅速整合性ディレクトリにアクセスし、データがプライベートとして保持されていることを取り出した情報が標示すればリコールを発信するステップ(c)と、メイン整合性ディレクトリにアクセスし、取り出した情報を使用して上記迅速整合性ディレクトリから取り出した上記情報を検査するステップ(d)と、要求されたデータを上記プロセッサへ提供し、上記迅速整合性ディレ

クトリを更新するステップ(e)と、を含む方法。  
(9) 上記ステップ(c)において上記リコールが見込みリコールであり、上記ステップ(d)において上記見込みリコールが失敗する場合訂正措置を始動する、上記(8)に記載の方法。

(10) 上記迅速ディレクトリが、上記迅速ディレクトリに記憶されたディレクトリ・データのコンテンツの関数として部分的に更新される、上記(8)に記載の方法。

【0086】

【発明の効果】本発明の迅速整合性ディレクトリの情報に従って出される見込みリコールの利用によって、ディレクトリに基づくキャッシュ整合性を持つ分散メモリ・マルチプロセッサ・システムの処理性能が向上する。

【図面の簡単な説明】

【図1】本発明に従ってディレクトリに基づくキャッシュ整合性を取り入れた分散メモリ・マルチプロセッサ・コンピュータ・システムのブロック図である。

【図2】図1のシステムのメモリ、データ・キャッシュおよびディレクトリ・キャッシュに関するメモリ形式を示す図表である。

【図3】図1のシステムにおける読み取り演算を実行する方法の流れ図である。

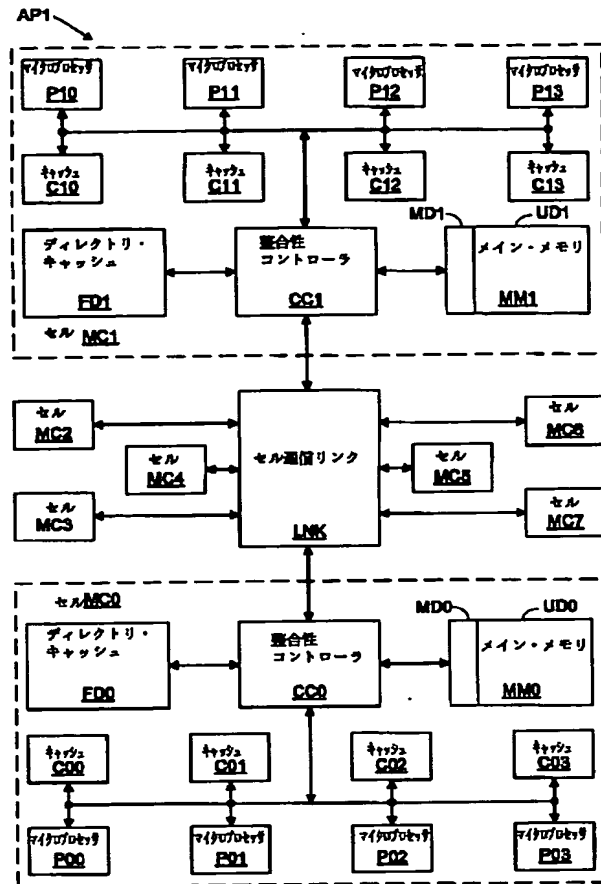
【符号の説明】

20 AP1 分散メモリ・マルチプロセッサ・システム  
MC0、MC1 メモリ・セル  
CC0、CC1 整合性コントローラ  
P00、P10 マイクロプロセッサ  
MM0、MM1 メイン・メモリ  
FD0、FD1 迅速ディレクトリ

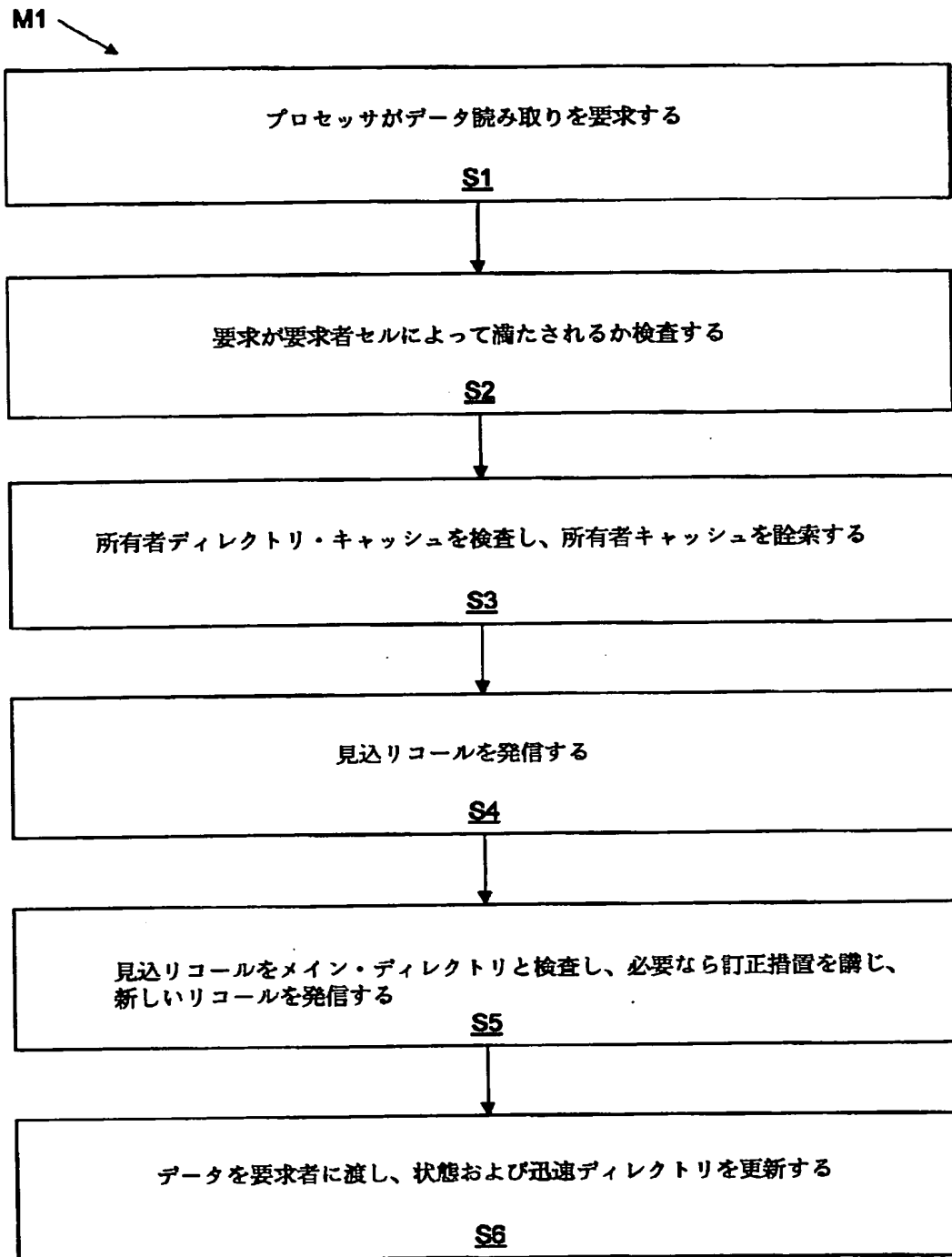
【図2】

メモリ形式					
メモリ	行アドレス/ インデックス	タグ	状態	サイト	ユーザ・ データ
メイン・メモリ (MDO)	28+3	0	2	8	512
ユーザ・データ・ キャッシュ (COO)	16	15	3	0	512
ディレクトリ・ キャッシュ (PDO) モードA	21	0	1	3	0
ディレクトリ・ キャッシュ モードB プライベート	20	1	0	3	0
ディレクトリ・ キャッシュ モードB パブリック	19	0	0	8	0

【図 1】



【図 3】



---

フロントページの続き

(72)発明者 ソリン・ラコボビッチ  
アメリカ合衆国95129カリフォルニア州サ  
ン・ノゼ、カウンテス・ドライブ 5990